

AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer system for building a health model of software components, comprising:

one or more processors;

system memory; and

one or more physical computer-readable media having stored thereon computer-executable instructions representing modules configured to build a health model of software components, the modules including:

an instrumentation collector for receiving information specifying instrumentation of software components;

a health model generator for creating a health model using the information about the instrumentation of the software components; and

an instrumentation analyzer for grouping instrumentation that result in the same transition from one state of the health model to another state of the health model, the instrumentation analyzer grouping mapped instrumentation into groups that have the same state of operation before the instrumentation is generated and the same state of operation after the instrumentation was generated.

2. (Original) The system of claim 1 further comprising a database for storing the information about the instrumentation of the software components.

3. (Original) The system of claim 1 further comprising a database for storing the health model.

4. (Original) The system of claim 1 wherein the health model comprises a state diagram with a transition from one state to another state for a group of instrumentation.

5. (Original) The system of claim 1 wherein the instrumentation collector comprises a spreadsheet for manual entry of information about instrumentation of software components.

6. (Original) The system of claim 1 wherein the instrumentation collector comprises a parser for automatically parsing a software component to extract information about instrumentation of the software component.

7. (Original) The system of claim 1 wherein the instrumentation analyzer comprises an application that groups the instrumentation events by filtering the instrumentation based upon the state of the software component before the occurrence of instrumentation and the state of the software component after the occurrence of instrumentation.

8. (Original) The system of claim 1 wherein the health model generator comprises an application that generates a state diagram.

9. (Currently Amended) A computer ~~readable~~storage medium having computer-executable components comprising the system of claim 1.

10. (Currently Amended) A method for building a health model of a software component, comprising the steps of:

receiving an inventory of instrumentation of the software component;

mapping the inventory of instrumentation to states of operation of the software component;

analyzing the inventory to ~~group~~ identify instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component; and

grouping the identified instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component, the grouping comprising grouping mapped instrumentation into groups that have the same state of operation before the instrumentation is generated and the same state of operation after the instrumentation was generated; and

generating the health model with the states of operation and at least one transition representing a group of instrumentation from one state of the health model to another state of the health model.

11. (Original) The method of claim 10 further comprising the step of creating an inventory of instrumentation of the software component.

12. (Original) The method of claim 11 wherein the step of creating an inventory of instrumentation of the software component comprises parsing the software component to extract information about instrumentation of the software component.

13. (Original) The method of claim 10 further comprising the step of determining states of operation of the software component.

14. (Original) The method of claim 13 wherein the step of determining states of operation of the software component comprises determining a stopped state.

15. (Original) The method of claim 13 wherein the step of determining states of operation of the software component comprises determining a running state.

16. (Original) The method of claim 13 wherein the step of determining states of operation of the software component comprises determining a failed state.

17. (Original) The method of claim 10 further comprising the step of adding instrumentation where there is none to indicate an occurrence of a transition from a failed state of operation to a running state of operation.

18. (Original) The method of claim 10 further comprising the step of adding instrumentation where there is none to indicate an occurrence of a transition from a running state of operation to a failed state of operation.

19. (Original) The method of claim 10 further comprising the step of persistently storing the inventory of instrumentation.

20. (Original) The method of claim 10 further comprising the step of persistently storing the generated health model.

21. (Original) The method of claim 10 further comprising revising the instrumentation of the software component.

22. (Original) The method of claim 21 further comprising updating the health model using the revised instrumentation.

23. (Original) The method of claim 22 further comprising generating a new health model.

24. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises receiving an inventory of one or more events.

25. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises receiving an inventory of one or more performance counters.

26. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises receiving an inventory of one or more error messages.

27. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises parsing the software component to extract information about instrumentation of the software component.

28. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises manually entering instrumentation information in a spreadsheet application.

29. (Original) The method of claim 10 wherein the step of analyzing the inventory comprises determining the state of operation before an instrumentation event occurs and the state of operation after the instrumentation event occurs.

30. (Original) The method of claim 10 wherein the step of analyzing the inventory to group instrumentation comprises filtering the instrumentation based upon the state of the software component before the occurrence of instrumentation and the state of the software component after the occurrence of instrumentation.

31. (Original) The method of claim 10 wherein the step of analyzing the inventory to group instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component comprises labeling each group of instrumentation as a single transition action from one state of operation of the software component to another state of operation of the software component.

32. (Original) The method of claim 10 wherein the step of analyzing the inventory to group instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component comprises using an application to analyze the inventory to group instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component.

33. (Original) The method of claim 10 wherein the step of analyzing the inventory comprises determining a component to blame for instrumentation indicating a failure of the software component.

34. (Original) The method of claim 10 wherein the step of generating the health model comprises generating a state diagram.

35. (Original) The method of claim 26 wherein the step of generating a state diagram comprises using an application to generate the state diagram.

36. (Currently Amended) A computer ~~readable-storage~~ medium having computer-executable instructions for performing the method of claim 10.

Claims 37-40. (Cancelled)

41. (New) The method of claim 10, wherein the health model is configured to detect cycles of change in states of operation.

42. (New) The method of claim 41, wherein at least one of the cycles of change in states of operation comprises a cycle of failure and recovery.

43. (New) At a computer system, a method for notifying a computer user when a software component transitions from one state of a health model to another state of the health model, the method comprising:

an act of instrumentation monitoring a software component to detect when a change of the state of operation occurs such that the operation is no longer running;

upon detecting a state transition that results in the software component no longer functioning as intended, an act of the instrumentation notifying the computer system with an alert, the alert indicating that the software component is no longer functioning as intended; and

upon subsequently detecting a state transition that results in the software component again functioning as intended, an act of the instrumentation notifying the computer system with an anti-alert, the anti-alert indicating that the software component is again functioning as intended.

44. (New) The method of claim 42, wherein the computer system automatically suppresses notification of a failure when an anti-alert is received.